This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1.   **(Currently amended)** A method for automatically generating all or part of an
Extensible Stylesheet Language Transforms ("XSLT") transform for transforming Extensible
Markup Language ("XML") data in a source file format into data in a new file format,
comprising:

producing an input file that identifies at least one data pattern from an XML
source file; and

producing a first ~~process~~ object for generating an XSLT transform, wherein
said first ~~process~~ object generates at least one first feature of ~~an output~~ said XSLT transform,
and wherein said first ~~process~~ object is designed to ~~incorporate~~ inherit a second ~~process~~
object for generating ~~an~~ said XSLT transform; and

incorporating said second ~~process~~ object into said first ~~process~~ object, wherein
said second ~~process~~ object uses said input file to generate at least one second feature of said
~~output~~ XSLT transform.

2.   **(Original)** The method of claim 1, wherein said input file identifies at least one data
pattern from an XML source file by identifying at least one XPath expression.

3.   **(Original)** The method of claim 1, wherein said input file also identifies at least one
output data format for a new file.

4.   **(Original)** The method of claim 3, further comprising pairing in said input file at least
one XPath expression used to identify the at least one data pattern to an output data format
for a new file.

5.   **(Currently amended)** The method of claim 1, further comprising using said first
~~process~~ object to override a call initiated by said second ~~process~~ object with a call to a portion
of said first ~~process~~ object for generating said at least one first feature of an output XSLT
transform.

6.   **(Currently amended)** The method of claim 1, wherein said second ~~process~~ object is an XSLT transform.

7.   **(Currently amended)** The method of claim 1, wherein said second ~~process~~ object includes a ~~process~~ object for placing at least one ~~prefabricated~~ custom XSLT transform into said output XSLT transform.

8.   **(Currently amended)** The method of claim 1, wherein said second ~~process~~ object generates at least one XSLT template corresponding to the at least one data pattern from an XML source file.

9.   **(Currently amended)** The method of claim 8, wherein said second ~~process~~ object includes a call to ~~an abstract named~~ a stub XSLT template that can be used by said first ~~process~~ object to initiate the insertion of additional features into said at least one XSLT template.

10.   **(Original)** The method of claim 1, wherein said input file conforms to an XML schema.

11.   **(Currently amended)** A system comprising a computer readable medium bearing data structures and instructions for generating XSLT transforms, comprising:

a first data structure comprising an input file containing at least one XPath expression; and

a first XSLT transform comprising instructions for generating at least one first feature of an output XSLT transform; ~~wherein said first XSLT transform incorporates~~ and instructions for incorporating a second XSLT transform, said second XSLT transform comprising instructions for generating at least one second feature of an output XSLT transform; and

a ~~process~~ object in said second XSLT transform, said object comprising instructions for generating an XSLT template or a portion thereof based on said at least one XPath expression.

12.   (Currently amended) The system of claim 11, wherein the input file conforms to an XML schema designed for ~~interoperability with~~ <u>readability by</u> said second XSLT transform.

13.   (Currently amended) The system of claim 11, further comprising ~~prefabricated~~ <u>custom</u> transforms <u>comprising instructions</u> that can be incorporated into said output XSLT transform.

14.   (Original) The system of claim 11, further comprising at least one identification of a new file data format in said input file.

15.   (Currently amended) The system of claim 11, further comprising a ~~process~~ <u>object</u> in said first XSLT transform <u>comprising instructions</u> to override a call to ~~an abstract named~~ <u>a</u> <u>stub</u> template initiated by said second XSLT transform with a call to a portion of said first XSLT transform ~~for generating said at least one first feature of an output XSLT transform~~.

16.   (Currently amended) The system of claim 11, further comprising a ~~process~~ <u>object</u> in said second XSLT transform <u>comprising instructions</u> to call ~~an abstract named~~ <u>a stub</u> XSLT template.

17.   (Currently amended) ~~An~~ <u>A computer readable medium comprising computer readable instructions, said instructions comprising an</u> XSLT transform, comprising:

        a first component <u>comprising instructions</u> for transforming at least one section of an input file into an XSLT template or portion thereof; and

        a second component <u>comprising instructions</u> to call to ~~an abstract named~~ <u>a stub</u> XSLT template.

18.   (Original) The XSLT transform of claim 17, further comprising a third component for identifying at least one namespace prefix from the input file and generating a header for an output XSLT transform including said namespace prefix.

19.   (Original) The XSLT transform of claim 18, further comprising a fourth component for generating a temporary namespace and inserting it into said header for an output XSLT transform.

20.   (Currently amended) The XSLT transform of claim 17, further comprising a ~~third~~ component for inserting at least one ~~prefabricated~~ custom XSLT template into an output XSLT transform.

21.   (Currently amended) A computer comprising means for automatically generating all or part of an Extensible Stylesheet Language Transforms ("XSLT") transform for transforming Extensible Markup Language ("XML") data in a source file format into data in a new file format, comprising:

      means for reading an input file that identifies at least one data pattern from an XML source file; and

      means for generating at least one first feature of an output XSLT transform with a first ~~process~~ object; and

      means for incorporating a second ~~process~~ object for generating an XSLT transform into said first ~~process~~ object; and

      means for said second ~~process~~ object to use said input file to generate at least one second feature of said output XSLT transform.

22.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said input file identifies at least one data pattern from an XML source file by identifying at least one XPath expression.

23.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said input file also identifies at least one data pattern for a new file.

24.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 23, further comprising means for pairing in said input file at least one XPath expression used to identify the at least one data pattern to an output data format for a new file.

25.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, further comprising means for using said first ~~process~~ object to override a call initiated by said second ~~process~~ object with a call to a portion

of said first ~~process~~ object for generating said at least one first feature of an output XSLT transform.

26.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said second ~~process~~ object is an XSLT transform.

27.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said second ~~process~~ object includes a means for referencing at least one ~~prefabricated~~ custom XSLT transform in said output XSLT transform.

28.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said second ~~process~~ object generates at least one XSLT template corresponding to the at least one data pattern from an XML source file.

29.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 28, wherein said second ~~process~~ object includes a call to ~~an abstract named~~ a stub XSLT template that can be used by said first ~~process~~ object to initiate the insertion of additional features into said at least one XSLT template.

30.   (Currently amended) The computer comprising means for automatically generating all or part of an XSLT transform of claim 21, wherein said input file conforms to an XML schema.